AMENDMENTS TO THE CLAIMS

1.-2. (Canceled)

3. (Currently amended) A method ~~as recited in Claim 2 wherein generating the updated key value further comprises:~~ of automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, the method comprising the computer-implemented steps of:

receiving a location value that identifies a location of the segment within the keystream;

generating the updated key value corresponding to the identified segment and based on a current key value by performing:

creating and storing values in a memory that correspond to a logical tree, wherein the tree represents the keystream, wherein each leaf node of the tree represents a particular keystream segment associated with a discrete location in the keystream, and wherein an order of each leaf node in pre-order traversal of the tree corresponds to a sequential order of all keystream segments;

creating and storing an ordered plurality of data elements, wherein each of the data elements is identifiable by a node value that is associated with a unique leaf node or intermediate node in the tree, and wherein each of the data elements stores a keystream segment;

advancing to the updated key by outputting that particular keystream segment stored in a next data element from among the ordered plurality of data elements, and updating the ordered plurality of data elements to discard used data elements;

seeking a future key without discarding the updated key by locating a highest internal node in the tree that is an ancestor node of that node corresponding to the updated key and applying one or more pseudo-random functions to each node in the tree that is traversed from the highest internal node to a node representing the future key.

4. (Currently amended) A method as recited in Claim 3, wherein ~~generating the updated key value further~~ seeking comprises generating the key value based on the steps of:

selecting a highest-ordered element from among the plurality of data elements;
if the selected element is not associated with a leaf node, then:
storing, in a new highest-ordered element among the plurality of data elements, a first new key value that is determined by applying a first pseudo-random function to the selected element;
generating a second new key value by applying a second pseudo-random function to the selected element.

5. (Currently amended) A method as recited in Claim 43, wherein ~~generating the updated key value~~ advancing further comprises generating the updated key value based on the steps of:
returning, as the updated key value, a segment of the keystream associated with the node identified in the selected next node value when such node is a leaf node;
returning, as the key value that is generated, a segment of the keystream associated with the second new node value when the node identified in the selected next node value is not a leaf node.

6. (Original) A method as recited in Claim 3, wherein generating the updated key value further comprises generating the key value based on the steps of:
determining a current location value;
identifying an internal node of the tree having a highest node number that is an ancestor of a first node corresponding to the received location value and of a second node corresponding to the current location value;
determining a path from the identified internal node to the first node;
traversing the path while applying a first pseudo-random key updating function to the then-current key value during each leftward downward transition and applying a second pseudo-random function during each rightward downward transition.

7. (Original) A method as recited in Claim 6, wherein generating the updated key value further comprises generating the updated key value based on the steps of:

storing, in a new highest-ordered element among the plurality of data elements, each new key value that is generated as part of applying the first and second pseudo-random functions;

generating, as the updated key value, the new key value that stored in the highest-ordered element among the plurality of data elements, when the first node is reached in traversing the path.

8. (Currently amended) A method as recited in Claim ~~2~~3, wherein each edge of the tree is associated with a distinct pseudo-random function that, when applied to a current key, results in generating a new updated key.

9. (Currently amended) A method as recited in Claim ~~2~~3, wherein each edge leading leftward and downward from a first node to a second node is associated with a first pseudo-random key updating function, and wherein each edge leading rightward and downward from the first node to a third node is associated with a second pseudo-random key updating function.

10. (Currently amended) A method as recited in Claim ~~6~~9, wherein each of the pseudo-random functions receives, as input, a first keystream segment and generates, as output, a second keystream segment based on updating the first keystream segment in a pseudo-random manner, such that determining the first keystream segment based on the second keystream segment is computationally infeasible.

11. (Currently amended) A method as recited in Claim ~~1~~3, further comprising the steps of distributing the updated key value to each member of a multicast group for use in secure communications among the multicast group.

12. (Currently amended) A method of automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, the method comprising the computer-implemented steps of:

receiving a location value that identifies a location of the segment within the keystream;

generating the updated key value corresponding to the identified segment and based on a current key value with forward security ~~and without relying on a key management process for providing such forward secrecy;~~ security, wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible~~;~~, by performing:

creating and storing values in a memory that correspond to a logical tree, wherein the tree represents the keystream, wherein each leaf node of the tree represents a particular keystream segment associated with a discrete location in the keystream, and wherein an order of each leaf node in pre-order traversal of the tree corresponds to a sequential order of all keystream segments;

creating and storing an ordered plurality of data elements, wherein each of the data elements is identifiable by a node value that is associated with a unique leaf node or intermediate node in the tree, and wherein each of the data elements stores a keystream segment;

selecting a highest-ordered element from among the plurality of data elements;

if the selected element is not associated with a leaf node, then:

storing, in a new highest-ordered element among the plurality of data elements, a first new key value that is determined by applying a first pseudo-random function to the selected element;

generating a second new key value by applying a second pseudo-random function to the selected element;

returning, as the updated key value, a segment of the keystream associated with the node identified in the selected next node value when such node is a leaf node;

returning, as the key value that is generated, a segment of the keystream associated with the second new node value when the node identified in the selected next node value is not a leaf node;

determining a current location value;

identifying an internal node of the tree having a highest node number that is an ancestor of a first node corresponding to the received location value and of a second node corresponding to the current location value;

determining a path from the identified internal node to the first node;

traversing the path while applying a first pseudo-random key updating function to the then-current key value during each leftward downward transition and applying a second pseudo-random function during each rightward downward transition;

storing, in a new highest-ordered element among the plurality of data elements, each new key value that is generated as part of applying the first and second pseudo-random functions;

generating, as the updated key value, the new key value that stored in the highest-ordered element among the plurality of data elements, when the first node is reached in traversing the path.

13.-18. (Canceled)

19. (Currently amended) An apparatus for automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, comprising:

means for receiving a location value that identifies a location of the segment within the keystream;

means for generating the updated key value ~~corresponding to the identified segment and based on a current key value with forward security and without relying on a key management process for providing such forward secrecy; and wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible.~~, comprising:

means for creating and storing values in a memory that correspond to a logical tree, wherein the tree represents the keystream, wherein each leaf node of the tree represents a particular keystream segment associated with a discrete location in the keystream, and wherein an order of each leaf node in pre-order traversal of the tree corresponds to a sequential order of all keystream segments;

means for creating and storing an ordered plurality of data elements, wherein each of the data elements is identifiable by a node value that is associated with a unique leaf

node or intermediate node in the tree, and wherein each of the data elements stores a keystream segment;

means for advancing to the updated key by outputting that particular keystream segment stored in a next data element from among the ordered plurality of data elements, and updating the ordered plurality of data elements to discard used data elements;

means for seeking a future key without discarding the updated key by locating a highest internal node in the tree that is an ancestor node of that node corresponding to the updated key and applying one or more pseudo-random functions to each node in the tree that is traversed from the highest internal node to a node representing the future key.

20. (Currently amended) An apparatus for automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, comprising:

a network interface that is coupled to the data network for receiving one or more packet flows therefrom;

a processor;

one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

receiving a location value that identifies a location of the segment within the keystream;

generating the updated key value corresponding to the identified segment and based on a current key value ~~with forward security and without relying on a key management process for providing such forward secrecy; and wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible~~, by performing:

creating and storing values in a memory that correspond to a logical tree, wherein the tree represents the keystream, wherein each leaf node of the tree represents a particular keystream segment associated with a discrete location in the keystream, and wherein an order of each leaf node in pre-order traversal of the tree corresponds to a sequential order of all keystream segments;

creating and storing an ordered plurality of data elements, wherein each of the data elements is identifiable by a node value that is associated with a unique leaf node or intermediate node in the tree, and wherein each of the data elements stores a keystream segment;

advancing to the updated key by outputting that particular keystream segment stored in a next data element from among the ordered plurality of data elements, and updating the ordered plurality of data elements to discard used data elements;

seeking a future key without discarding the updated key by locating a highest internal node in the tree that is an ancestor node of that node corresponding to the updated key and applying one or more pseudo-random functions to each node in the tree that is traversed from the highest internal node to a node representing the future key.

21.-22. (Canceled)

23. (Currently amended) An apparatus as recited in Claim ~~22~~19, wherein the means for ~~generating the updated key value~~ seeking further comprises means for selecting a highest-ordered element from among the plurality of data elements; means, if the selected element is not associated with a leaf node, for storing, in a new highest-ordered element among the plurality of data elements, a first new key value that is determined by applying a first pseudo-random function to the selected element, and for generating a second new key value by applying a second pseudo-random function to the selected element.

24. (Currently amended) An apparatus as recited in Claim ~~23~~19, wherein the means for generating the updated key value further comprises means for returning, as the updated key value, a segment of the keystream associated with the node identified in the selected next node value when such node is a leaf node; means for returning, as the key value that is generated, a segment of the keystream associated with the second new node value when the node identified in the selected next node value is not a leaf node.

25. (Currently amended) An apparatus as recited in Claim ~~22~~19, wherein the means for generating the updated key value further comprises ~~generating the key value based on the steps of~~:
means for determining a current location value;
means for identifying an internal node of the tree having a highest node number that is an ancestor of a first node corresponding to the received location value and of a second node corresponding to the current location value;
means for determining a path from the identified internal node to the first node;
means for traversing the path while applying a first pseudo-random key updating function to the then-current key value during each leftward downward transition and applying a second pseudo-random function during each rightward downward transition.

26. (Currently amended) An apparatus as recited in Claim 25, wherein the means for generating the updated key value further comprises means for storing, in a new highest-ordered element among the plurality of data elements, each new key value that is generated as part of applying the first and second pseudo-random functions; means for generating, as the updated key value, the new key value that stored in the highest-ordered element among the plurality of data elements, when the first node is reached in traversing the path.

27. (Currently amended) An apparatus as recited in Claim ~~21~~19, wherein each edge of the tree is associated with a distinct pseudo-random function that, when applied to a current key, results in generating a new updated key.

28. (New) An apparatus as recited in Claim 20, wherein the instructions for seeking further comprise instructions for performing:
selecting a highest-ordered element from among the plurality of data elements;
if the selected element is not associated with a leaf node, then:

storing, in a new highest-ordered element among the plurality of data elements, a first new key value that is determined by applying a first pseudo-random function to the selected element;

generating a second new key value by applying a second pseudo-random function to the selected element.

29. (New) An apparatus as recited in Claim 20, wherein the instructions for advancing further comprise instructions for performing:

returning, as the updated key value, a segment of the keystream associated with the node identified in the selected next node value when such node is a leaf node;

returning, as the key value that is generated, a segment of the keystream associated with the second new node value when the node identified in the selected next node value is not a leaf node.

30. (New) An apparatus as recited in Claim 20, wherein the instructions for generating the updated key value further comprise instructions for performing:

determining a current location value;

identifying an internal node of the tree having a highest node number that is an ancestor of a first node corresponding to the received location value and of a second node corresponding to the current location value;

determining a path from the identified internal node to the first node;

traversing the path while applying a first pseudo-random key updating function to the then-current key value during each leftward downward transition and applying a second pseudo-random function during each rightward downward transition.

31. (New) An apparatus as recited in Claim 20, wherein the instructions for generating the updated key value further comprise instructions for performing:

storing, in a new highest-ordered element among the plurality of data elements, each new key value that is generated as part of applying the first and second pseudo-random functions;

generating, as the updated key value, the new key value that stored in the highest-ordered element among the plurality of data elements, when the first node is reached in traversing the path.

32. (New) An apparatus as recited in Claim 20, wherein each edge of the tree is associated with a distinct pseudo-random function that, when applied to a current key, results in generating a new updated key.

33. (New) An apparatus as recited in Claim 20, wherein each edge leading leftward and downward from a first node to a second node is associated with a first pseudo-random key updating function, and wherein each edge leading rightward and downward from the first node to a third node is associated with a second pseudo-random key updating function.

34. (New) An apparatus as recited in Claim 33, wherein each of the pseudo-random functions receives, as input, a first keystream segment and generates, as output, a second keystream segment based on updating the first keystream segment in a pseudo-random manner, such that determining the first keystream segment based on the second keystream segment is computationally infeasible.

35. (New) An apparatus as recited in Claim 20, further comprising instructions for performing distributing the updated key value to each member of a multicast group for use in secure communications among the multicast group.